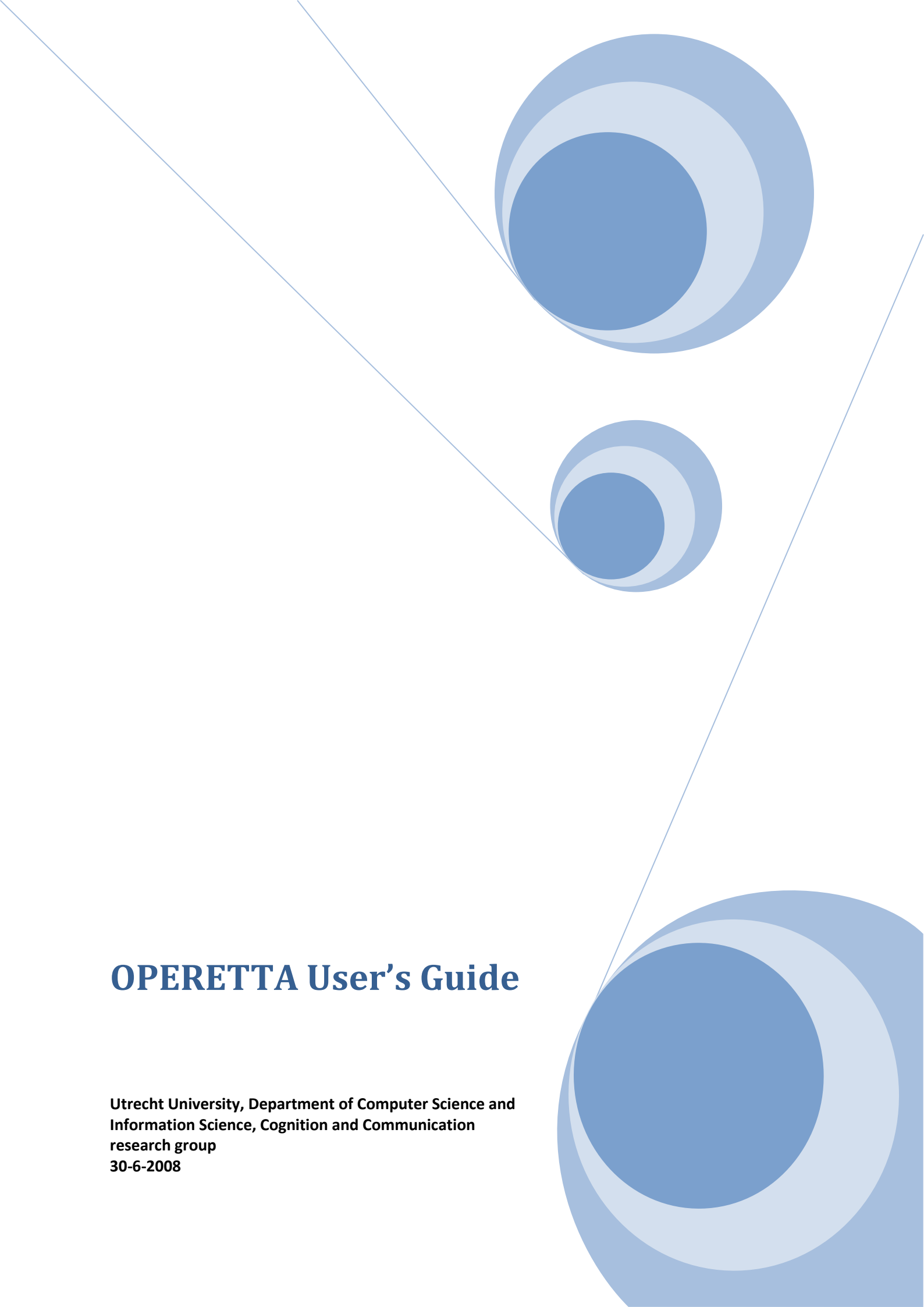




OPERETTA User's Guide

**Utrecht University, Department of Computer Science and
Information Science, Cognition and Communication
research group
30-6-2008**

Contents

1	Installation guide	3
1.1	Installing the OperettA modeling repository	3
1.2	Configuring RacerPro.....	4
2	Usage guide	6
2.1	Introduction.....	6
2.2	Selecting the OperettA modeling tools	7
3	Creating a project	8
3.1	Creating a graph: Interaction Structure View	10
3.1.1	Creating a scene	10
3.1.2	Scene details.....	11
3.1.3	Linking scenes together.....	14
3.2	Creating a graph: Role Dependency View	18
3.2.1	Creating a new role	19
3.2.2	Role details	20
3.2.3	Creating dependencies between roles.....	24
3.3	Associating an Interaction Structure to a Role Dependency graph	27
3.4	Data analysis.....	27
3.5	Examples.....	27

1 Installation guide

Operetta which is the MetaEdit+ implementation of the Opera Methodology is compressed in a Zip file (operetta.zip) that can be downloaded from Operetta Project web site:

<http://www.cs.uu.nl/research/projects/opera/>

Before you go into the next section you have to properly install MetaEdit+ in your computer. To obtain a free trial version of MetaEdit+ go to MetaCase Web site on <http://www.metacase.com/>. Different licenses are provided especially for education or nonprofit organization at symbolic prices. Also in order to use all functionality of operetta you must install RacerPro Descriptive Logic Reasoner which is free for research purpose: <http://www.racer-systems.com/>

When you install MetaEdit+4.5, after indicating the folder in which you want to install the program, you will be asked to specify the path to the folder you would like METAEDIT+ to use as his **“DATABASE folder”**, also called the **“root repository”** (cf. Figure 1), please choose it carefully and remember it, you will need it later (please, try to use a folder without white spaces in his path).

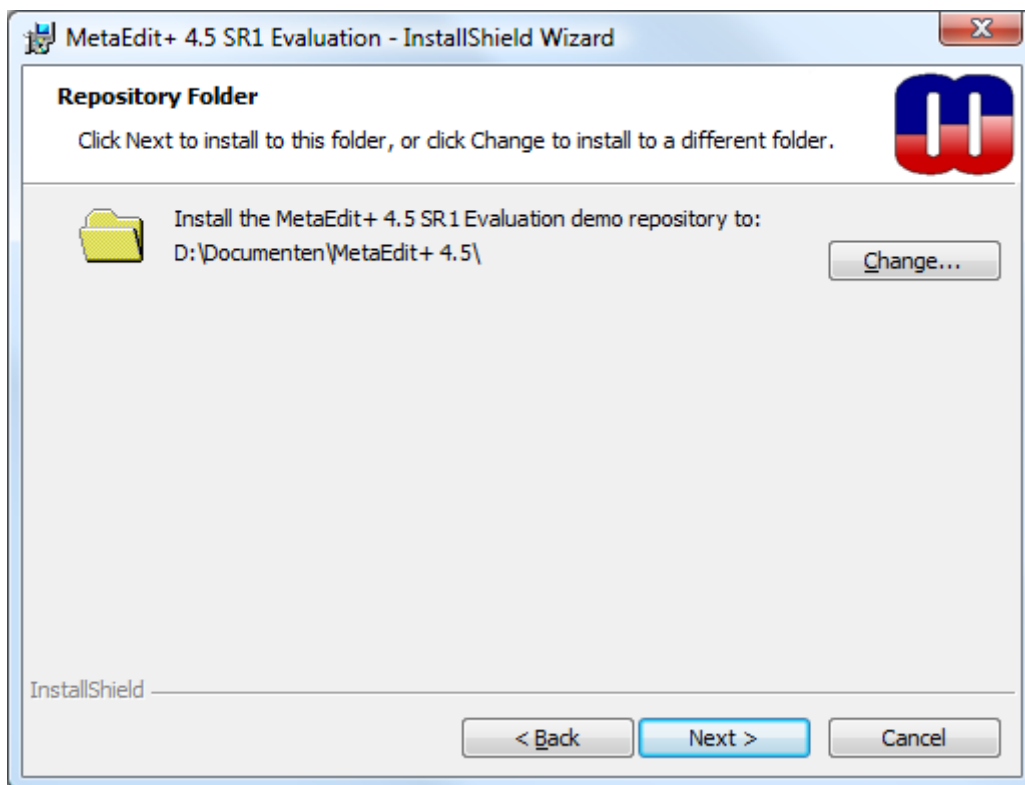
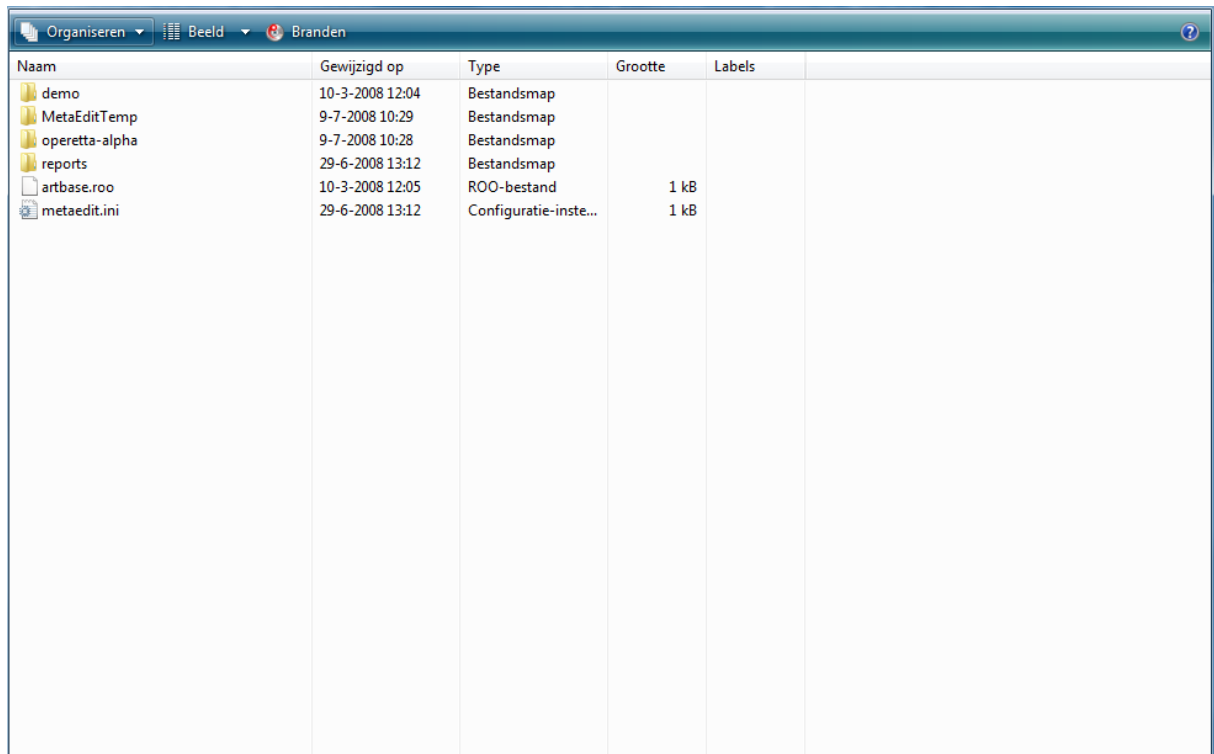


Figure 1: Specification of the Repository Folder

1.1 Installing the Operetta modeling repository

To install, Operetta, just go into the database folder (root repository) and unzip the downloaded Operetta zip-file inside that folder. Your folder should now look like presented in Figure 2.



Naam	Gewijzigd op	Type	Grootte	Labels
demo	10-3-2008 12:04	Bestandsmap		
MetaEditTemp	9-7-2008 10:29	Bestandsmap		
operetta-alpha	9-7-2008 10:28	Bestandsmap		
reports	29-6-2008 13:12	Bestandsmap		
artbase.roo	10-3-2008 12:05	ROO-bestand	1 kB	
metaedit.ini	29-6-2008 13:12	Configuratie-inste...	1 kB	

Figure 2: Contents of the Repository Folder

Metaedit+4.5 will automatically detect operetta-alpha folder which contains the meta-model and add it as a new repository and hence make operetta available. MetaEditTemp folder contains Third-parties elements associated to the functioning of Operetta.

1.2 Configuring RacerPro

After you have installed RacerPro anywhere on your computer, it is required that you add an environmental variable to your operating system. To do this, follow these steps:

- Step 1: Go to the System properties in the Control Panel. (Control Panel -> System in classic-view).
- **Vista only:** Click on "Change Settings".
- Step 2: Go to the "Advanced" tab.

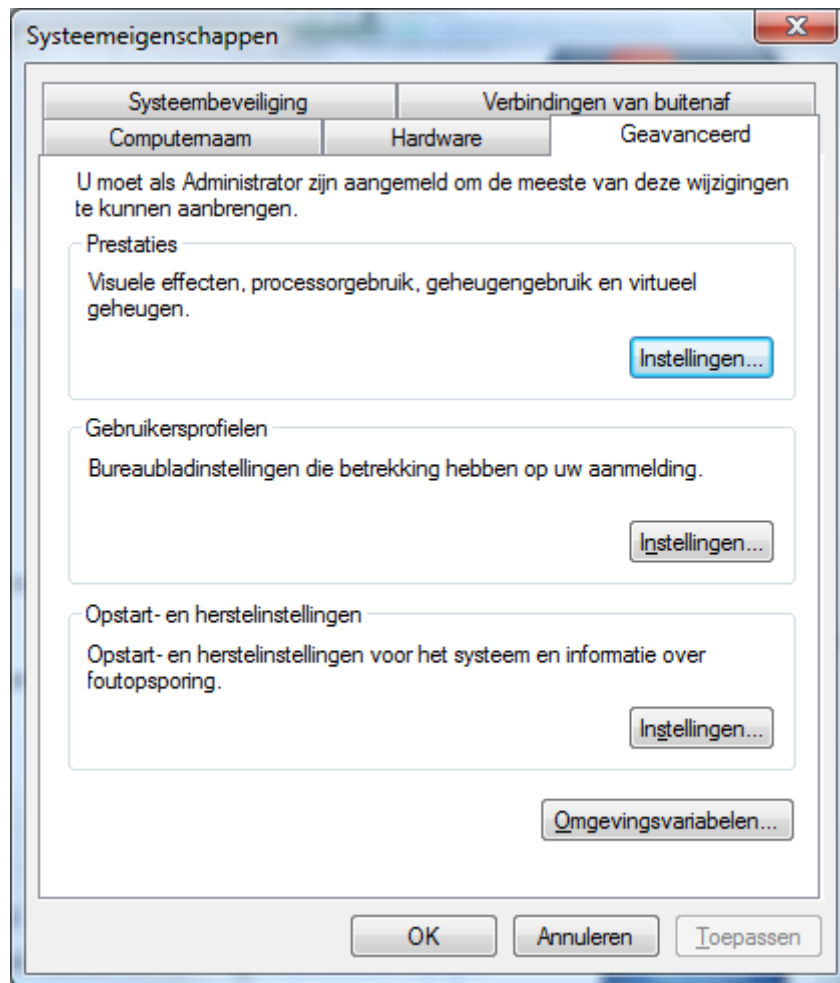


Figure 3: Advanced tab of System properties

- Step 3: Click on "Environment Variables"
- Step 4: Create a new System variable by clicking on the "New..." button.
- Step 5: Name the variable "METAEDIT_DATABASE".

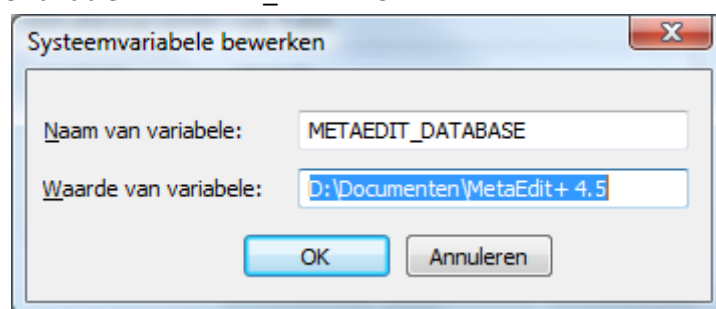


Figure 4: Creating a new system variable

- Step 6: The value of the variable should be the location of the **root repository**.
- Step 7: Press the "OK" button to continue (repeat).

2 Usage guide

2.1 Introduction

MetaEdit+ uses an object-oriented repository management strategy to store user's defined data. It means that all users' defined data, which include meta-models (or modeling languages) with their concepts and associated generators, as well as models and their constituents plus their properties, are stored and manage as object following the well defined object-oriented model. For that matter, the underlying object data model used by MetaEdit+ is **GOPRR**, which is an acronym from the words Graph-Object-Property-Port-Role-Relationship. In other terms, each time you will be creating models, MetaEdit will store and manage them as graph constituted of objects that have properties, linked trough relationship and so on. However we won't explain in details the data model here, if you wish further information, please refer to MetaEdit+ documentation.

A repository could be viewed as a shared workspace. It brings the advantage to carefully manage the sharing of data across, modeling language, models, tools, but also across multiple users. Hence, modification of system designs (models) or even modeling languages in one tool or MetaEdit+ client is reflected everywhere it appears, guaranteeing consistent and up to date information across projects, models or users etc... Similarly, code and document generation are based directly on information stored in the repository.

The database folder as mentioned earlier is the root repository. This repository which is meant to be managed by MetaEdit itself contains all the user's defined repositories as "operetta-version-beta-01" which in turn contains user's defined data (ex: models).

Therefore in order to use operetta you must have operetta modeling language information and all that comes with it (generators, third party libraries, etc...) in your repository. That is why we provide you with a ready to use repository "operetta-version-beta-01" containing all necessary information for running operetta.

Repositories are structure into projects. Those are area inside a repository, defined by the user to organize and store what he has created (models, meta-models, generators, etc...) as a result of his work. Indeed, each time a user wants to define and store data in a repository in needs to do it inside a project. However, it is important to notice that this separation doesn't mean unavailability of data between projects. Indeed the sharing (or availability) of data between projects, is ensured by the repository management strategy as explains above.

A simplified view summarizing the above explanation is given in Figure 5.

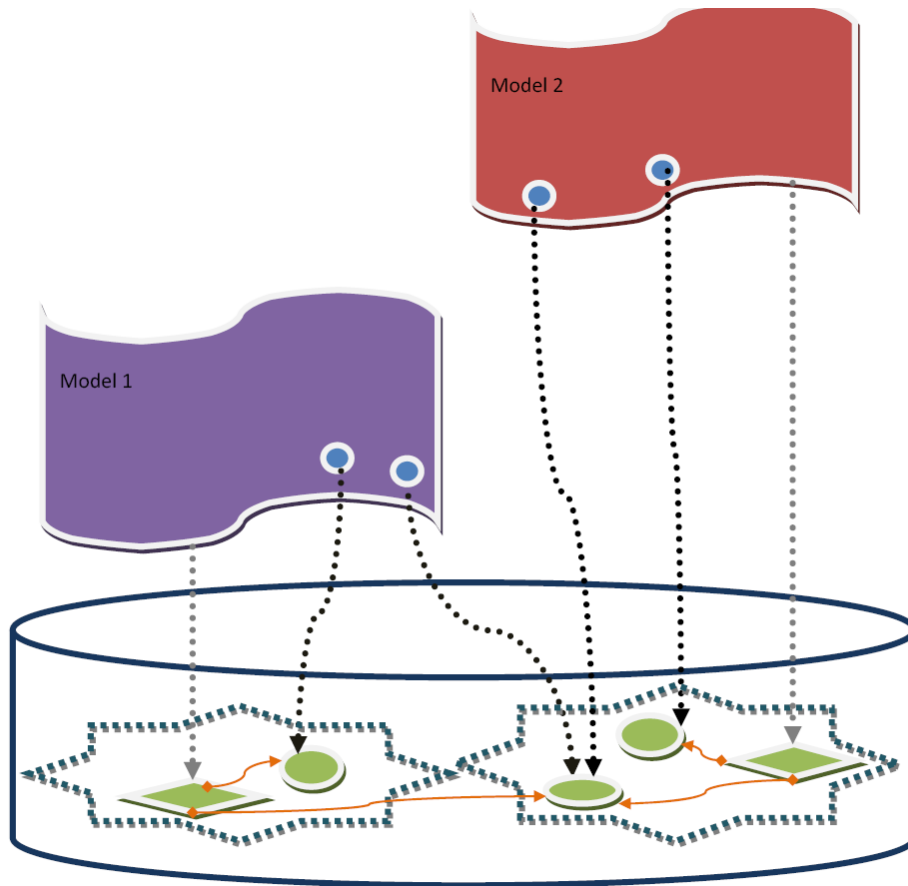


Figure 5: Graphical representation of a repository

In our case, we have defined one project inside “operetta-alpha” repository called “operetta” in which we have regrouped all the data that define the Operetta Editor. The following section will illustrate it further.

Bearded in mind with this knowledge we can move onto the next section. It gives a step by example on how to create multi-agent organization models with operetta.

2.2 Selecting the Operetta modeling tools

When you start Metaedit+4.5 you automatically launch the start-up-Launcher. Here you can see the different repositories available. You can choose the repository to connect to by selecting it with a left-click. Once selected as in Figure 6, you can see the projects (or areas) of the repository. It is possible to select some, all or none of them in order to be open while connecting to the repository. At the bottom of the window you can choose the type of user for the connection to the repository.

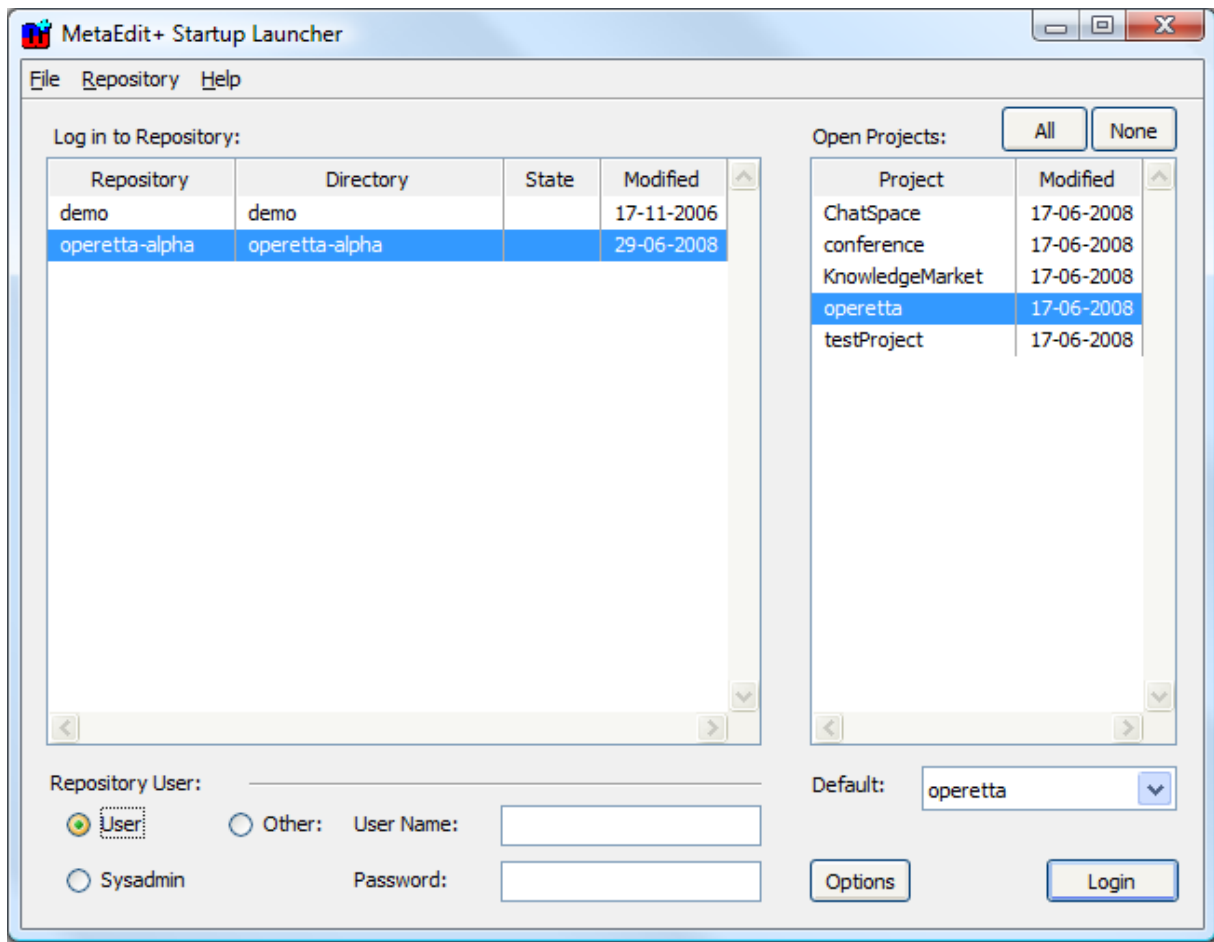


Figure 6: MetaEdit Startup Screen

- Step 1: Click on the operetta-alpha repository.
- Step 2: For the repository user (at the bottom), select “User”.
- Step 3: Click “None” to deselect all projects.
- Step 4: Click on the project called ‘operetta’ to highlight it.
- Step 5: Click on Login.

Once connected to a repository, MetaEdit+4.5 provide you with the Main-Launcher view (Figure 7). It brings you a well organize view of the repository from which you can launch any editor define in it, as well as generic tools provided by MetaEdit. In other terms anything necessary in order to access, inspect, create, modify or delete data.

3 Creating a project

Let’s now create our project in which we are going to store all our models.

- Step 1: Right-click on the projects panel (blank space) and choose “New...”.
- Step 2: Click “Yes” to continue.
- Step 3: Give a name for your new project (eg. “HousePainting”).
- Step 4: Click “Ok” when confirmed with the creation of the new project.
- Step 5: Set the default (lower-left corner) to your new project so that graphs will be added to the correct project.

If all went well you should have the situation represented in Figure 7. We should have two projects open now. Although we will only be adding graphs to our newly created project, the operetta project has to be open as well to allow us to use graph-types defined in this project.

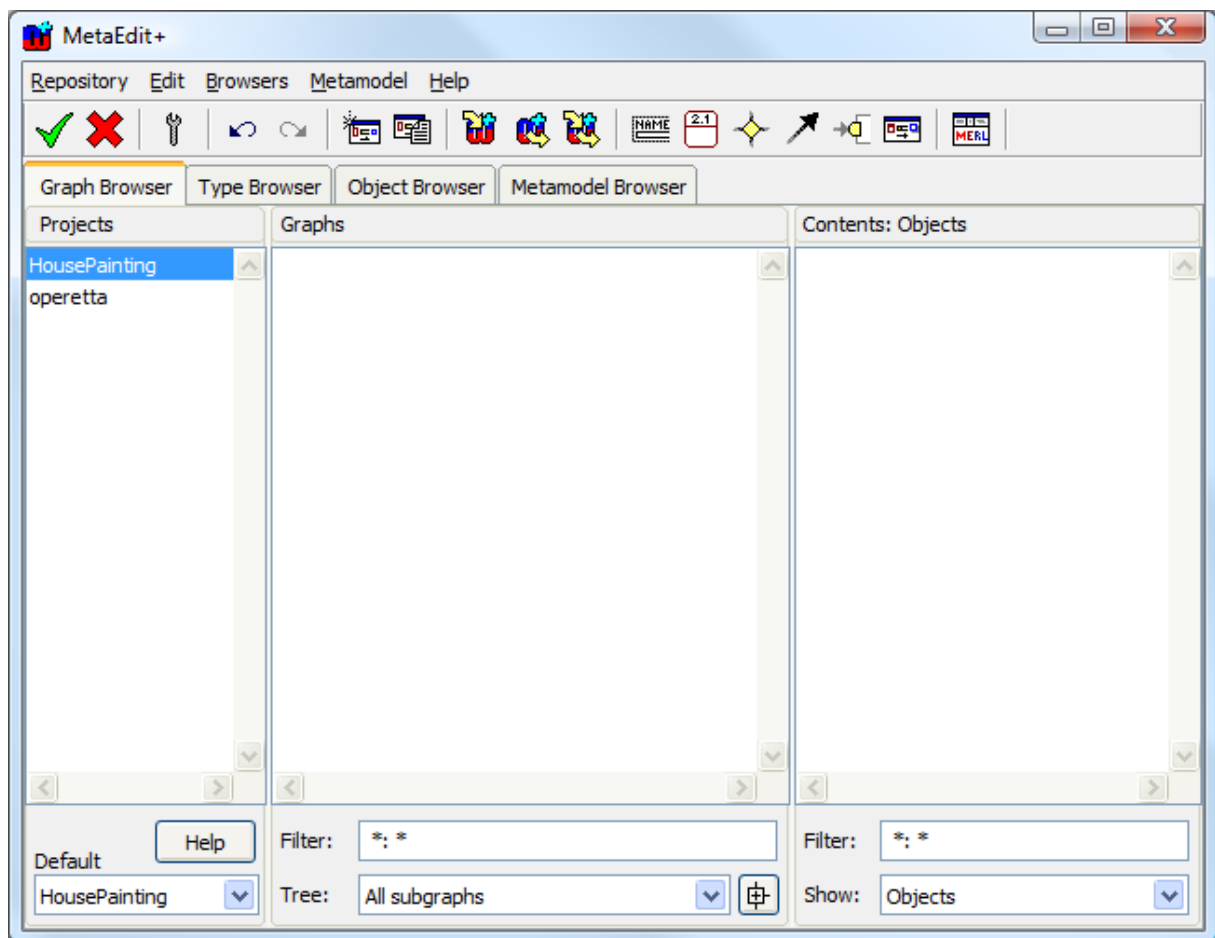


Figure 7: Project view after adding a new project

3.1 Creating a graph: Interaction Structure View

First we will create an Interaction Structure graph. To do this, follow these steps:

- Step 1: Right-click on the “Graphs” pane (blank area) and select “Create Graph...”
- Step 2: As the Graph type, select “Interaction structure View”.
- Step 3: Leave the “Open as” option on Diagram and click on “Ok” to continue.
- Step 4: Enter a name for your graph (e.g. PaintingInteractionStructure).
- Step 5: Click “Ok” to continue.

The Interaction Structure View window should open automatically (Figure 8). Here you can create your graph using the tools supplied on the second row of icons in the toolbar (e.g. STT).

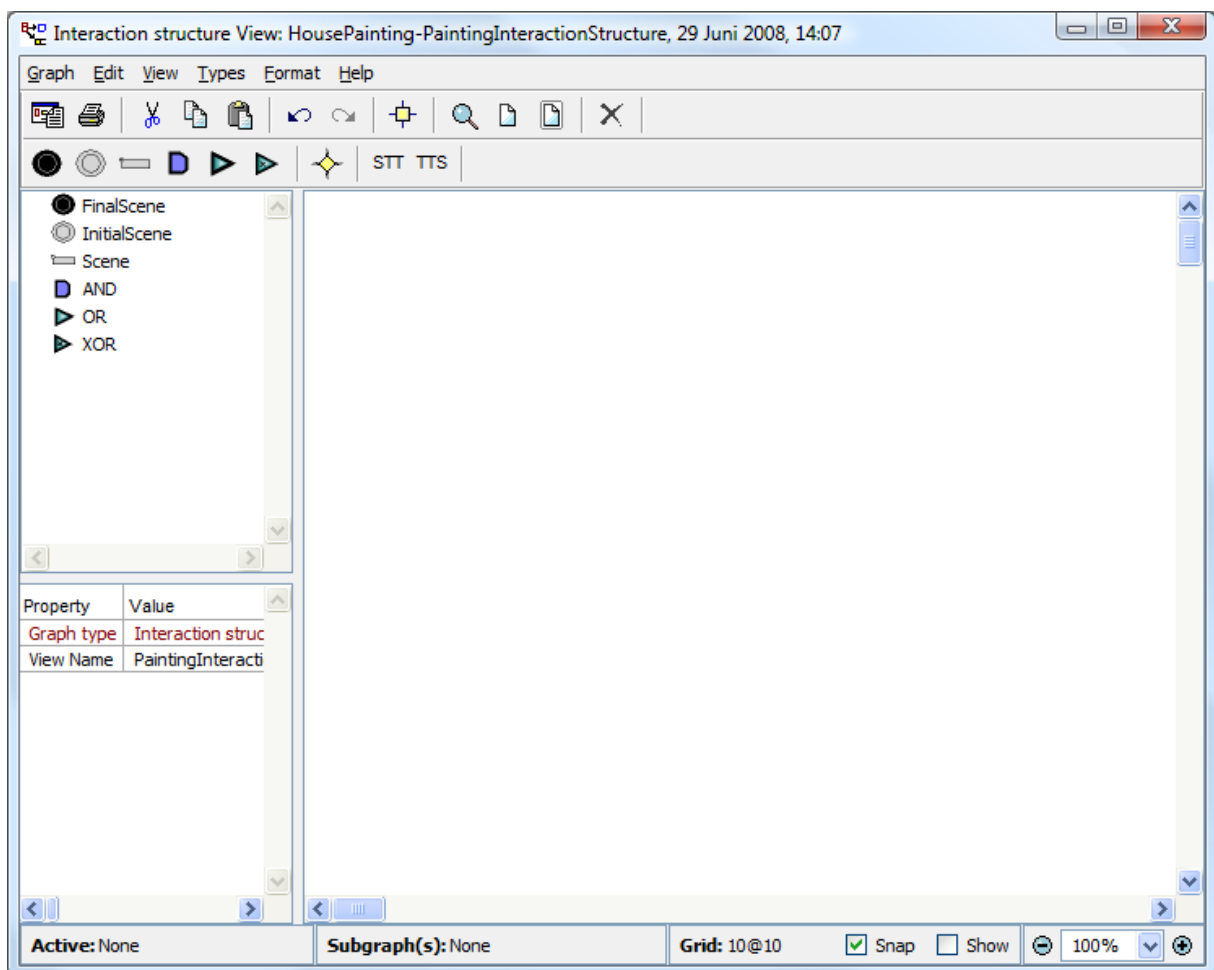


Figure 8: Interaction Structure View window

3.1.1 Creating a scene

To create a new scene, follow these steps:

- Step 1: Select the “Scene”-tool from the toolbar or from the menu (Types -> Scene). **Note: it is also possible to add special initial (Types -> InitialScene) and final (Types -> FinalScene) scenes to the Interaction Structure. The way to specify these is identical to any other scene.**
- Step 2: Click somewhere on the blank graph area where you want to create the new scene.
- Step 3: Fill in the details of the new scene (details below).
- Step 4: Click “OK” to create the new scene.

3.1.2 Scene details

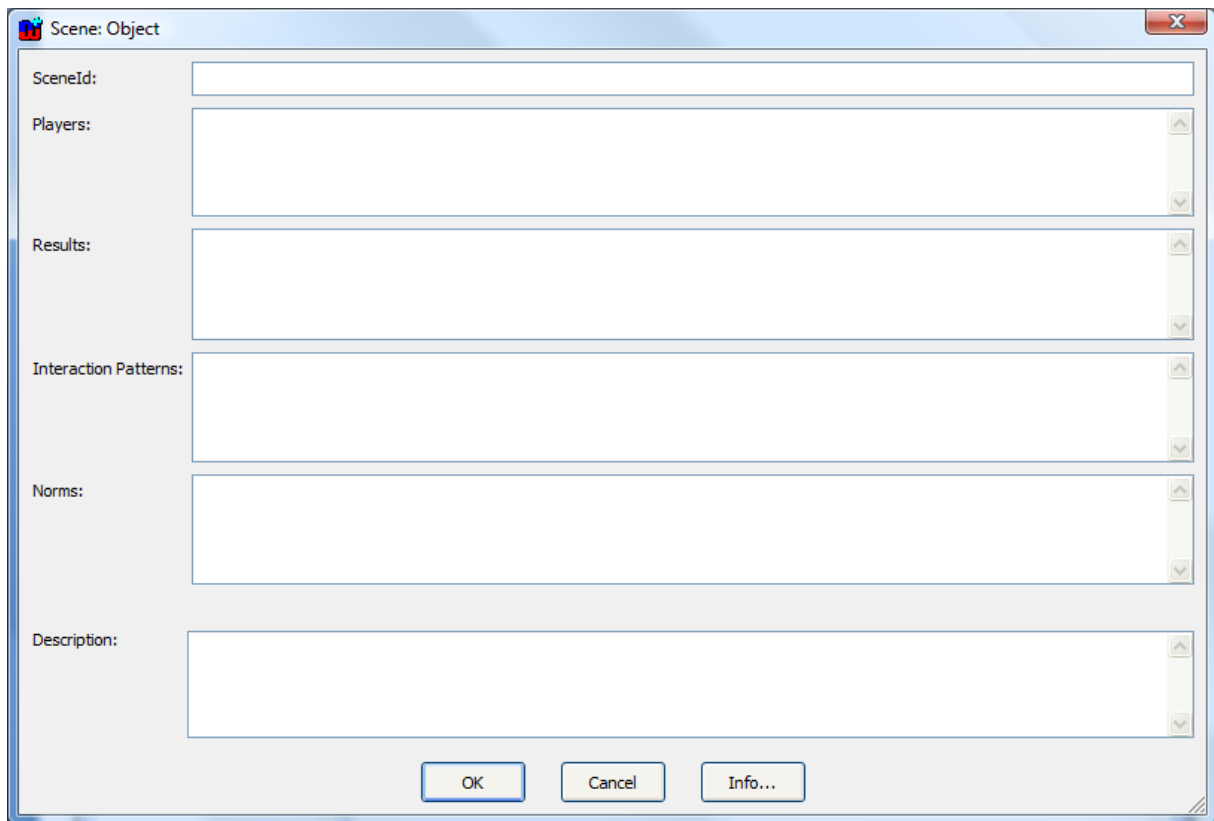


Figure 9: Scene details screen

A scene has several data that is associated with it (Figure 9):

- **Sceneld**
A name to identify this scene.
- **Players**
Roles that are associated with this scene.
- **Results**
The declarative expressions that describe the desired final states for the scene.
- **Interaction Patterns**
The description of the way to achieve a scene result.
- **Norms**
The description of the expected behavior of actors in a scene.
- **Description**
A description of the scene.

The **Sceneld** and **Description** can be filled in by providing the field with basic text. The other fields however require a different interaction method:

3.1.2.1 Players

It is possible to add players to a scene. It is advised to do this after you have created the necessary roles in a Role Dependency graph.

To add a player to the scene:

- Step 1: Right-click the Players pane and select “Add Element...”.

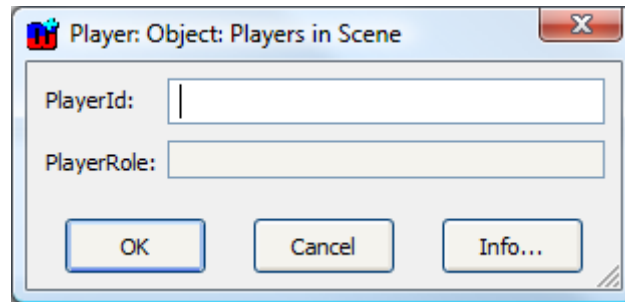


Figure 10: Adding Element to Players

- Step 2: Fill in an identifier for the Player (e.g. “Painter”).
- Step 3: Double-click on PlayerRole.

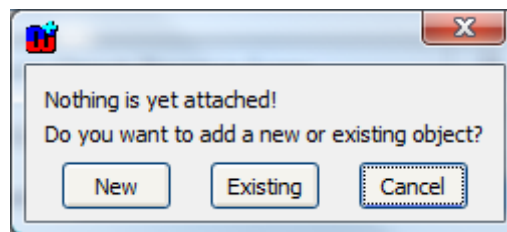


Figure 11: Attach a PlayerRole

- Step 4: If you have defined a role in a Role Dependency Graph, click “Existing”, select the appropriate role in the next screen and Press “Ok”.
- Step 5: Press “Ok” to add the Player to the scene.

3.1.2.2 Results

To specify the results of a scene:

- Step 1: Right-click the Results pane and select “Add Element...”.
- Step 2: Fill in an identifier for the Result (e.g. “r1”).

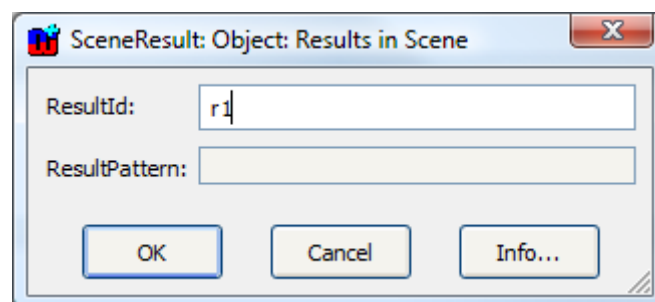


Figure 12: Adding results to a scene

- Step 3: Double-click on ResultPattern.
- Step 4: Click “New” to specify a new pattern.

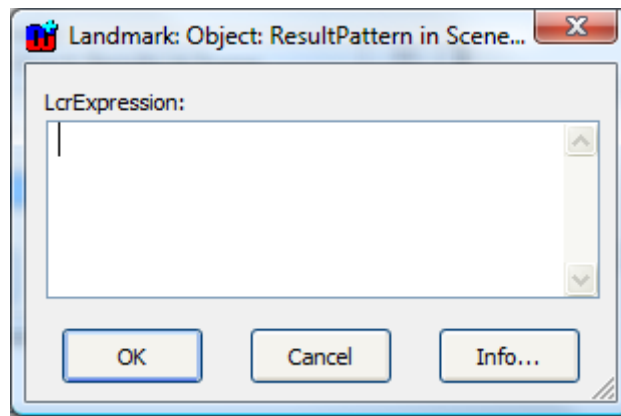


Figure 13: Specifying a new Result pattern

- Step 5: Fill in the pattern and click "Ok".
- Step 6: Click "Ok" to add the result to the scene.

3.1.2.3 Interaction Patterns

To define the interaction patterns of a scene:

- Step 1: Right-click the Interaction Patterns pane and select "Add Element...".

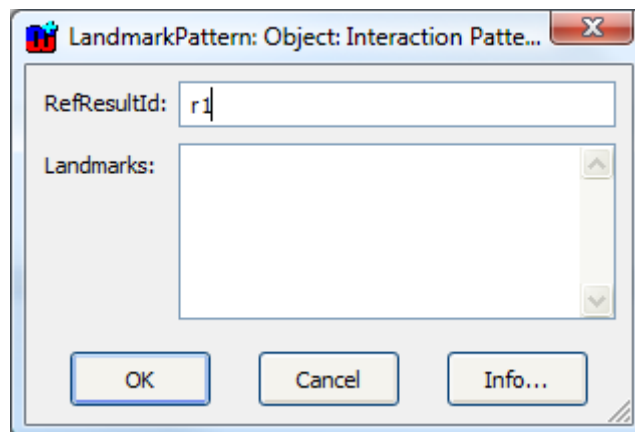


Figure 14: Adding an Interaction Pattern to a Scene

- Step 2: Fill in an identifier that corresponds with the a Result identifier that is associated with this interaction pattern (e.g. “r1”).
- Step 3: Add landmark(s):
 - Step 1: Right-click the Landmarks pane and select “Add Element...”.

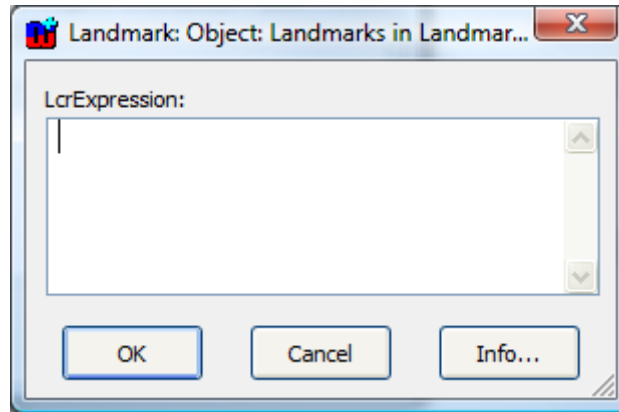


Figure 15: Specifying a landmark for an Interaction Pattern

- Step 2: Specify the expression that defines this landmark.
 - Step 3: Click “Ok” to add this landmark to the Interaction Pattern.
- Step 4: Click “Ok” to add the Interaction Pattern to the scene.

3.1.2.4 Norms

To specify the norms of a scene:

- Step 1: Right-click the Norms pane and select “Add Element...”.
- Step 2: Fill in the norm condition, detection mechanism, saction and repair fields.
- Step 3: Click on “Ok” to add the Norm to the scene.

3.1.3 Linking scenes together

It is possible to link scenes to each other by defining arcs and transitions between them. Note that for a one-to-one link to a scene the transition is arbitrary. To create a one-to-one link between two scenes(e.g. Figure 16):

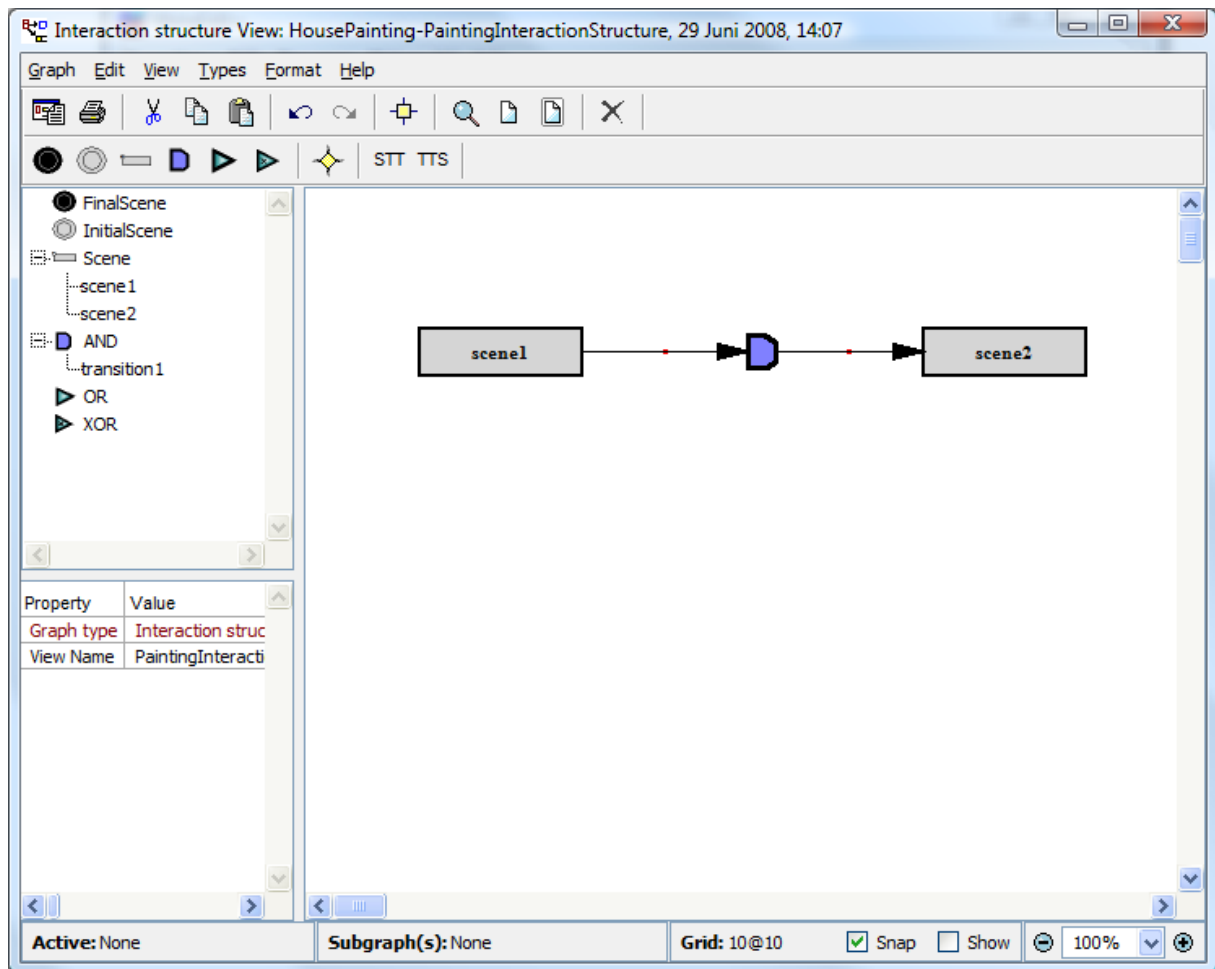


Figure 16: A one-to-one link between two scenes

- Step 1: Add a transition, such as AND by selecting it on the toolbar or in the menu (Types -> AND) and clicking on the structure view where you want to put it.
- Step 2: Define the transition name and roles evolutions:
 - o Step 1: Fill in the transition name.
 - o Step 2: Add role evolutions:
 - Step 1: Right-click on the roles evolutions pane.
 - Step 2: Click "Add Element..."

The screenshot shows a dialog box titled 'RoleEvolution: Object: Roles Evolutions in AND'. It contains the following fields and controls:

- Evolution Type:** A dropdown menu with 'necessary' selected.
- Incoming Role:** An empty text input field.
- Outgoing Role:** An empty text input field.
- Condition:** An empty text input field.
- At the bottom, there are three buttons: 'OK', 'Cancel', and 'Info...'.

Figure 17: Transition Role Evolutions

- Step 3: Specify the Evolution Type
 - Step 4: Specify the Incoming Role (double-click and choose “Existing”).
 - Step 5: Specify the Outgoing Role (double-click and choose “Existing”).
 - Step 6: Specify the Condition (double-click and choose “New”).
- Step 3: Click “Ok” to add a role evolution to the transition.
- Step 3: Add a Scene-To-Transition (STT):
 - Step 1: Click on STT in the toolbar or select it from the menu (Types -> SceneToTransitionArc).
 - Step 2: Click on the scene where the arc should originate from.
 - Step 3: Click on the transition where the arc should lead to.
- Step 4: Add a Transition-To-Scene (TTS):
 - Step 1: Click on STT in the toolbar or select it from the menu (Types -> TransitionToSceneArc).
 - Step 2: Click on the transition where the arc should originate from.
 - Step 3: Click on the scene where the arc should lead to.

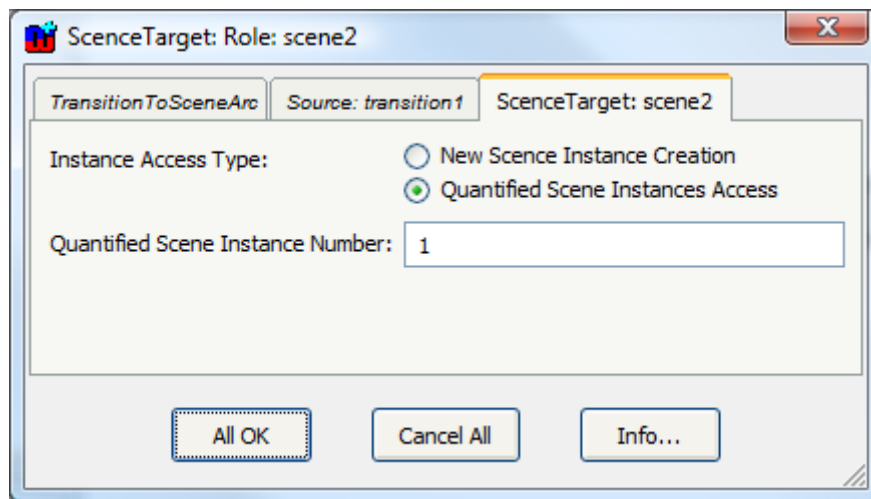


Figure 18: Properties of a Transition-To-Scene Arc

- Step 4: Specify the Instance Access Type
- Step 5: If the Instance Access Type is set to “Quantified Scene Instances Access”, specify the Instance Number.

Creating one-to-many or many-to-one links is done by adding multiple STT’s or TTS’s to the solution (e.g. in Figure 19).

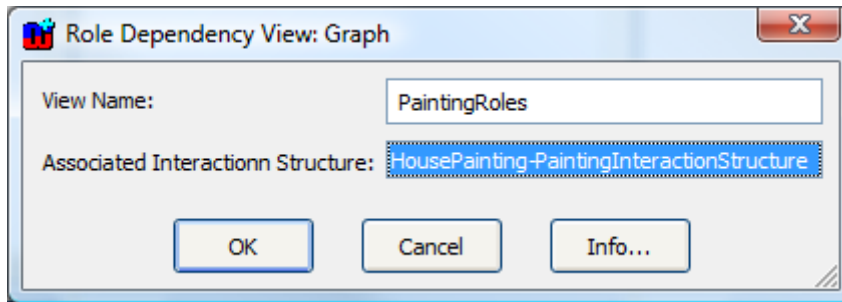


Figure 19: A one-to-many link between three scenes

3.2 Creating a graph: Role Dependency View

In this section we will discuss the Role Dependency graph. To create a new graph, follow these steps from the Main-Launcher view (Figure 7) :

- Step 1: Right-click on the “Graphs” pane (blank area) and select “Create Graph...”
- Step 2: As the Graph type, select “Role Dependency View”.
- Step 3: Leave the “Open as” option on Diagram and click on “Ok” to continue.



- Step 4: Enter a name for your graph (e.g. PaintingRoles).
- Step 5: To associate an Interaction Structure with this Role Dependency View:
 - o Step 1: Double-click the area right next to “Associated Interaction Structure”.
 - o Step 2: Click “Existing” and choose the Interaction Structure you wish to associate with this Role Dependency graph.
 - o Step 3: Click “OK” to associate this graph with the new Role Dependency graph.
- Step 5: Click “Ok” to continue.

The Role Dependency View window should open automatically (). Here you can create your graph using the tools supplied on the second row of icons in the toolbar (e.g. Depe) and do the analysis using the “OWL” and “SPA” buttons.

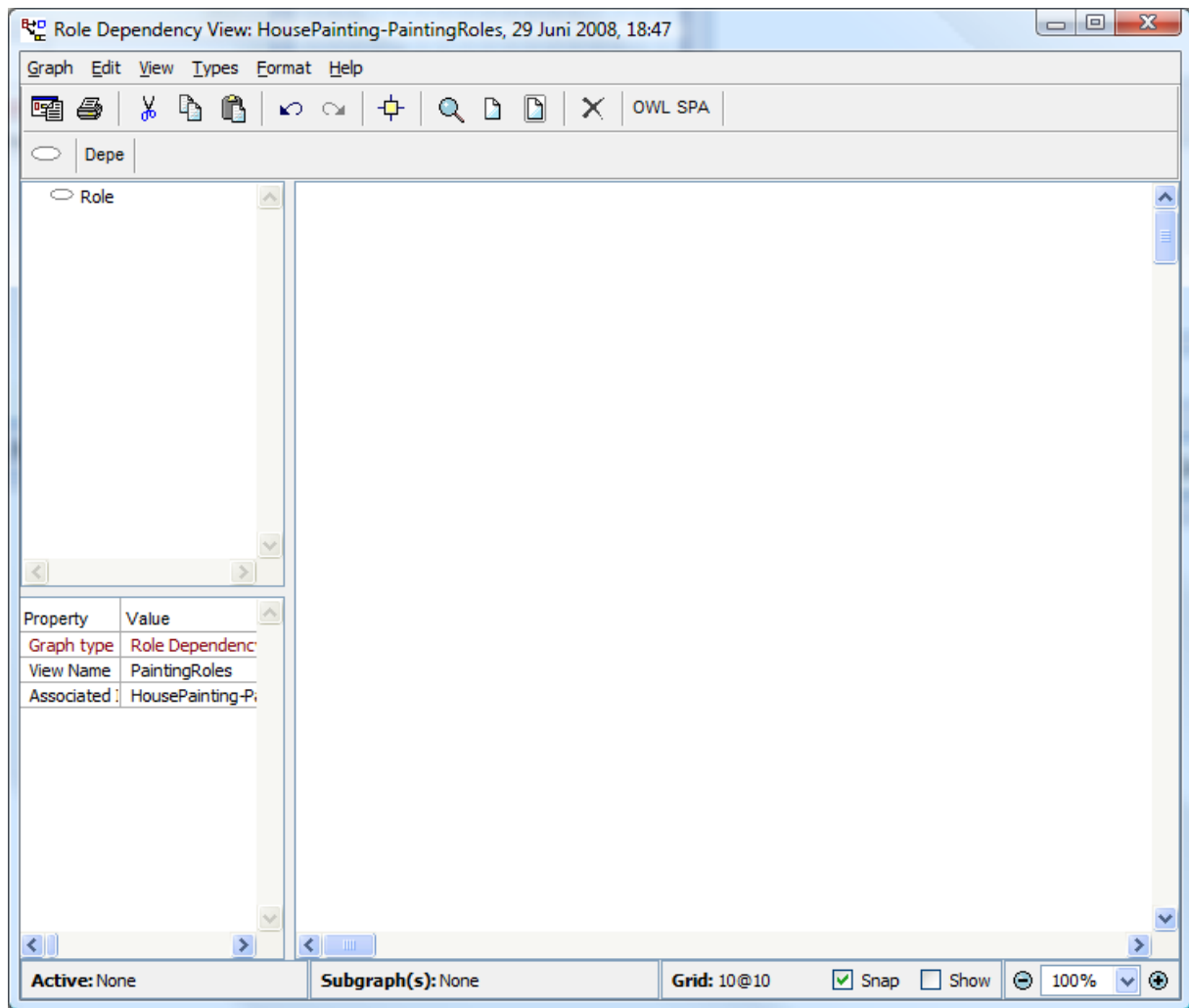


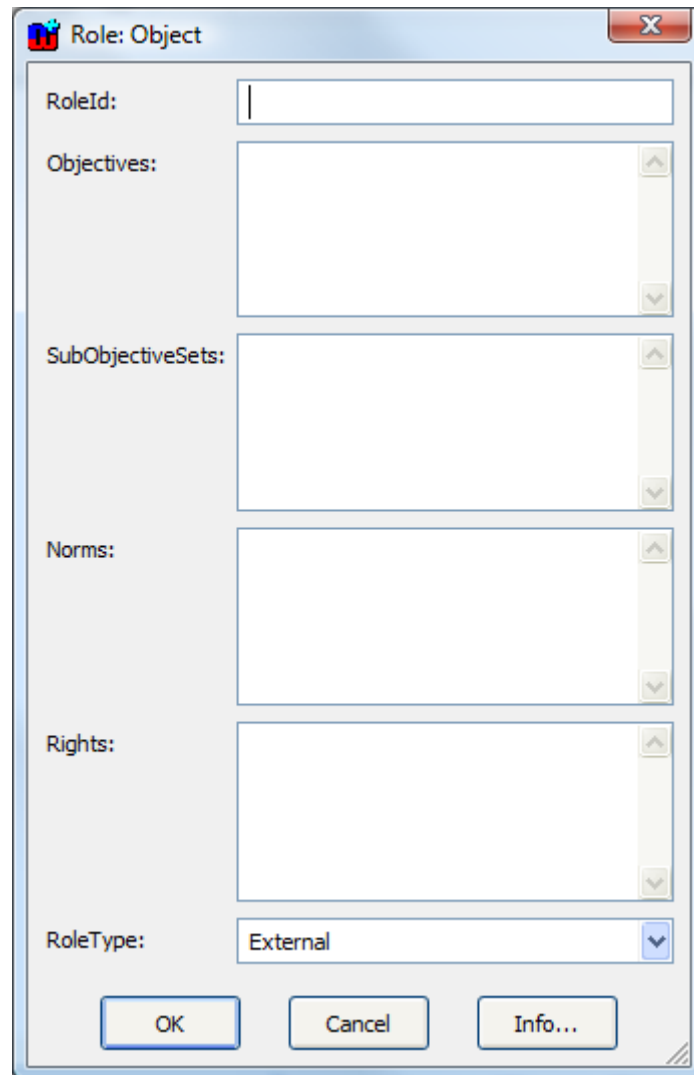
Figure 20: Role Dependency View

3.2.1 Creating a new role

To create a new role, follow these steps:

- Step 1: Click on the new role button, or use the menu (Types -> Role).
- Step 2: Click on the graph area where you want to place this new role.
- Step 3: Fill in the fields associated with a role (details below).
- Step 4: Click "OK" to add the new role.

3.2.2 Role details



The image shows a software dialog box titled "Role: Object". It contains several input fields and a list of buttons at the bottom. The fields are: "RoleId:" with a text input box; "Objectives:" with a list box; "SubObjectiveSets:" with a list box; "Norms:" with a list box; "Rights:" with a list box; and "RoleType:" with a dropdown menu currently showing "External". The buttons at the bottom are "OK", "Cancel", and "Info...".

Figure 21: Role details

A role has several data that is associated with it (Figure 21):

- **RoleId**
The name of this role which identifies it.
- **Objectives**
Role objectives are defined as states of affairs expected to be achieved in the environment.
- **SubObjectiveSets**
The sum of a set sub-objectives is considered to describe a referenced objective in more detail.
- **Norms**
Role norms specify the rules of behavior for actors performing that role, irrespective of the interaction scene.
- **Rights**
Role rights indicate the capabilities that actors of the role receive when enacting the role.
- **RoleType**
Actors of internal roles are fixed and controlled by the society and are designed to enforce

the social behavior of other agents and to assure the global activity of the society. External roles can in principle be enacted by any agent, according to the access rules specified by the society and describe the overall objectives of the society.

The **Sceneld** can be filled in by providing the field with basic text. The **RoleType** can be either external or internal. How to change the other fields is discussed in detail below.

3.2.2.1 Objectives

To create a new objective for a role, follow these steps:

- Step 1: Right-click the objectives area and choose “Add Element...”.

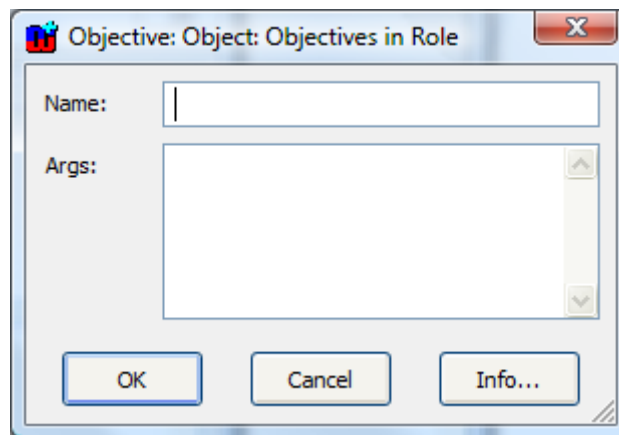


Figure 22: Add objective

- Step 2: Specify a name for the new objective.
- Step 3: To add arguments to the objective:
 - o Step 1: Right-click the Args area and choose “Add Element...”.

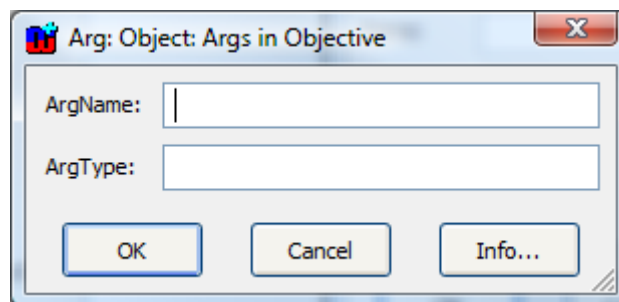


Figure 23: Defining arguments for an objective

- o Step 2: Specify a name and a type for the argument
 - o Step 3: Click on “OK” to add the argument to the objective.
- Step 4: Click “OK” to add the objective to the role.

3.2.2.2 SubObjectiveSets

To add a (set of) sub-objective(s), follow these steps:

- Step 1: Right-click the SubObjectiveSets area and choose “Add Element...”.

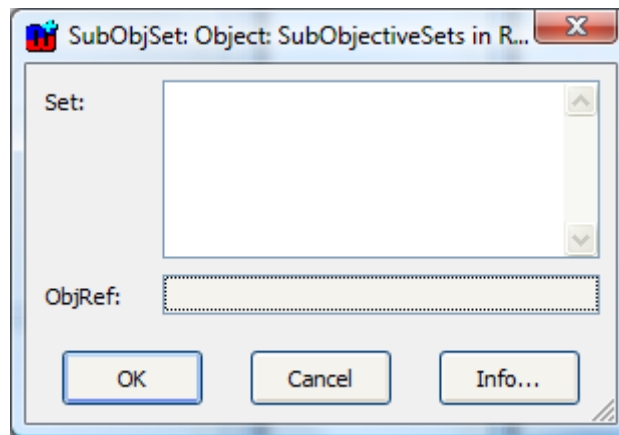


Figure 24: Add SubObjectiveSet

- Step 2: Add sub-objectives to the set:
 - o Step 1: Right-click on the Set area and choose “Add Element...”

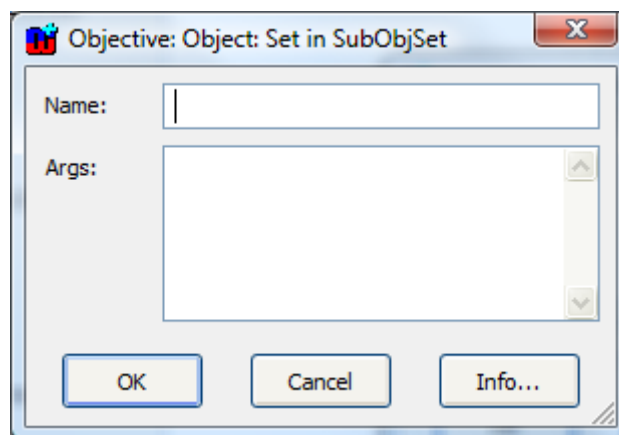


Figure 25: Add subobjective to set

- o Step 2: Specify a name for the new sub-objective.
- o Step 3: To add arguments to the sub-objective:
 - Step 1: Right-click the Args area and choose “Add Element...”.

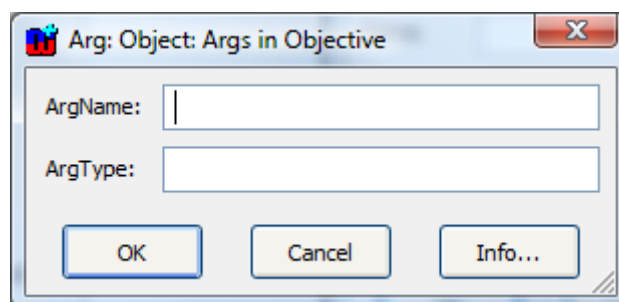


Figure 26: Add argument to subobjective

- Step 2: Specify a name and a type for the argument
- Step 3: Click on “OK” to add the argument to the sub-objective.
- Step 4: Click “OK” to add the sub-objective to the set of sub-objectives.
- Step 2: Reference to the objective this sub-objective set should be associated with in the area next to “ObjRef” by:
 - Step 1: Right-click the area next to ObjRef and choose “Attach Existing Object”.
 - Step 2: Double-click on the objective that should be associated with this set.
 - Step 3: Click “OK” to associate this objective with the new sub-objective set.
- Step 3: Click “OK” to add the sub-objective set to the role.

3.2.2.3 Norms

To specify the norms of a role:

- Step 1: Right-click the Norms pane and select “Add Element...”.

The image shows a standard Windows-style dialog box. The title bar at the top reads "Norm: Object: Norms in Role" and includes a close button (X). The main area of the dialog is divided into four sections, each with a label and a corresponding text input field with a vertical scrollbar:

- NormCondition:** The first text input field.
- DetectionMechanism:** The second text input field.
- Sanction:** The third text input field.
- Repair:** The fourth text input field.

At the bottom of the dialog box, there are three buttons: "OK", "Cancel", and "Info...".

Figure 27: Adding a norm to a role

- Step 2: Fill in the norm condition, detection mechanism, saction and repair fields ().
- Step 3: Click on “Ok” to add the Norm to the role.

3.2.2.4 Rights

- Step 3: Add landmark(s):
 - Step 1: Right-click the Landmarks pane and select “Add Element...”.

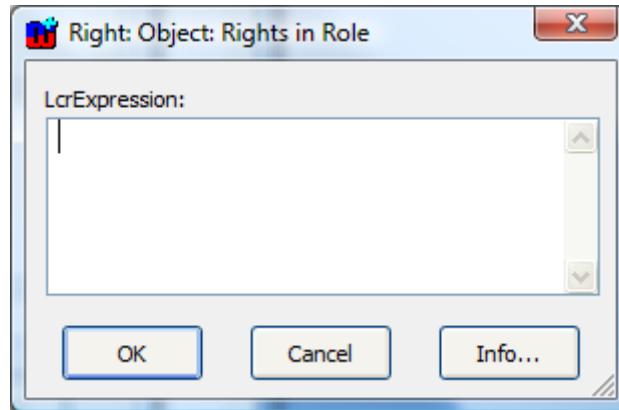


Figure 28: Specifying a right of a role

- Step 2: Specify the expression that defines this right.
- Step 3: Click “Ok” to add this right to the role.

3.2.3 Creating dependencies between roles

One role can be dependent on another role to succeed in doing its objective (e.g. Figure 29). To specify this, follow these steps:

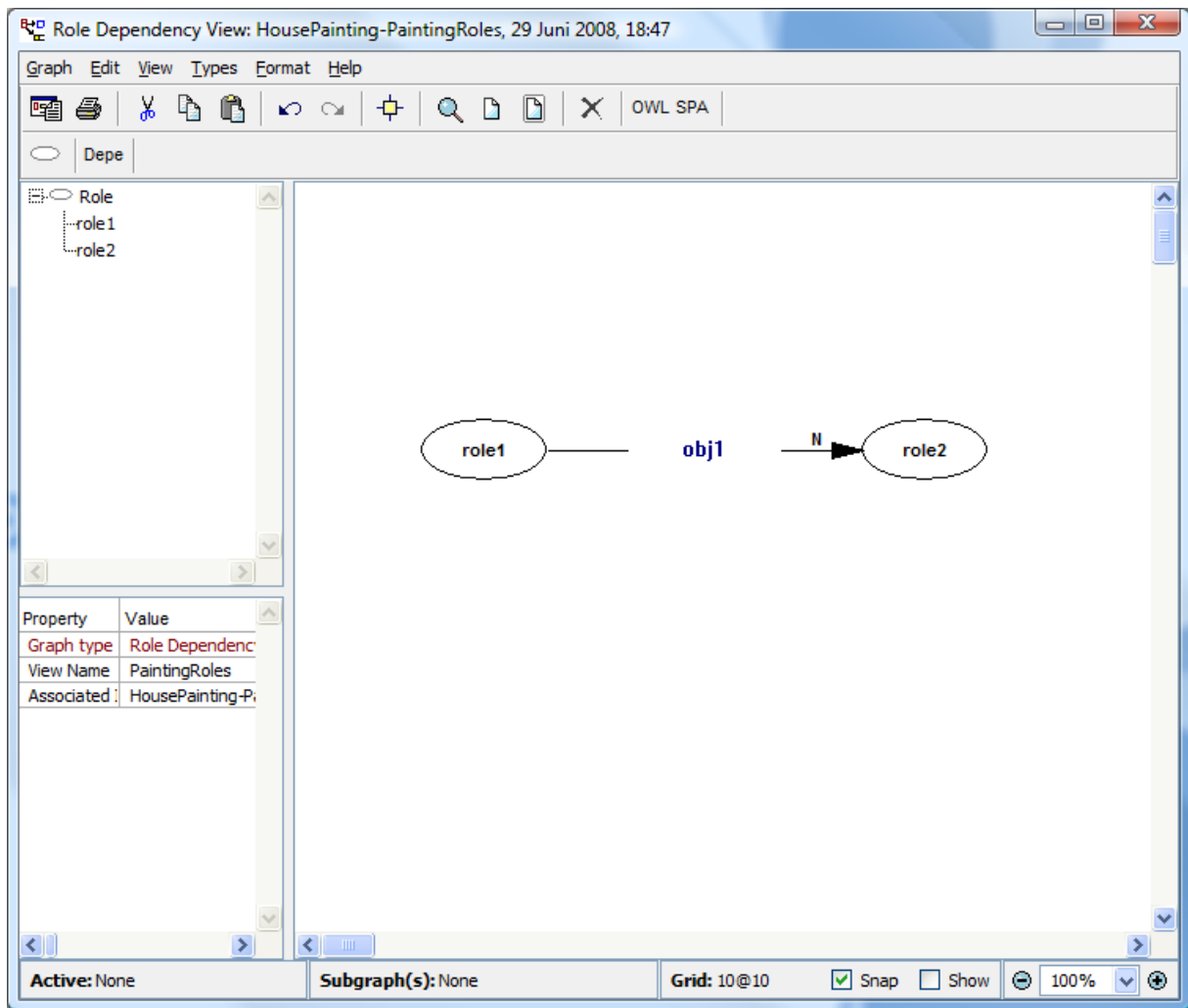


Figure 29: Two roles with a network dependency between them

- Step 1: Click on the “Depe” tool on the toolbar or use the menu (Types -> Dependency).
- Step 2: Click on the dependent role.
- Step 3: Click on the role it depends on.

Dependency: Relationship

Dependency Depender: role1 Dependee: role2

ObjectOfDependency:

Relation Type: Network

realisation scene:

All OK Cancel All Info...

Figure 30: Adding a dependency to the graph

- Step 4: In the pop-up window, define the properties of this relationship:
 - Step 1: Specify the objective that is the object of this dependency:
 - Step 1: Right-click on the area beside “ObjectOfDependency” and choose “Attach Existing Object...”.
 - Step 2: Choose the objective that is the object of dependency.
 - Step 3: Click “OK” to associate this objective with the relationship.
 - Step 2: Choose the relation type (network, market or hierarchy)
 - Step 3: Specify a scene where this dependency is realized:
 - Step 1: Right-click the area besides “realization scene” and choose “Add Existing...”.
 - Step 2: Double-click on the scene you wish to associate with this dependency.
 - Step 3: Click “OK” to associate the scene with this dependency.
 - Step 4: Click “All OK” to add the dependency to the graph.

3.3 Associating an Interaction Structure to a Role Dependency graph

If you already created a role dependency graph but didn't supply an associated Interaction Structure you can follow these steps to associate an interaction structure:

- Step 1: In the Role Dependency View, click on some blank space in the graph area.
- Step 2: In the properties window (lower-left corner), double-click "View Name".
- Step 3: To associate an Interaction Structure with this Role Dependency View:
 - o Step 1: Double-click the area right next to "Associated Interaction Structure".
 - o Step 2: Click "Existing" and choose the Interaction Structure you wish to associate with this Role Dependency graph.
 - o Step 3: Click "OK" to associate this graph with the new Role Dependency graph.

3.4 Data analysis

In the Role Dependency View you can use the analyse buttons ("OWL" and "SPA") to generate and view the corresponding OWL representation or to run an analysis of your model or respectively. However In order to run the analysis, you must have RacerPro Descriptive Reasoner running. So before Launch RacerPro, and once launched, click on SPA.

3.5 Examples

Example graphs can be found in the projects "ChatSpace" and "conference". Open these projects like you would your own to view the graphs.